WEST

Generate Collection | Print

**Search Results -** Record(s) 1 through 4 of 4 returned.

1. Document ID: US 6174595 B1

L1: Entry 1 of 4 File: USPT Jan 16, 2001

US-PAT-NO: 6174595
DOCUMENT-IDENTIFIER: US 6174595 B1

TITLE: Composites under self-compression

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Image

2. Document ID: US 5780806 A

L1: Entry 2 of 4 File: USPT Jul 14, 1998

US-PAT-NO: 5780806
DOCUMENT-IDENTIFIER: US 5780806 A

TITLE: Laser ablation system, and method of decontaminating surfaces

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Image

3. Document ID: US 5136528 A

L1: Entry 3 of 4 File: USPT Aug 4, 1992

US-PAT-NO: 5136528
DOCUMENT-IDENTIFIER: US 5136528 A

TITLE: Maintenance and operational simulators

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Image

4. Document ID: US 5111413 A

L1: Entry 4 of 4 File: USPT May 5, 1992

US-PAT-NO: 5111413
DOCUMENT-IDENTIFIER: US 5111413 A

TITLE: Computer-aided engineering

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | KWIC
Draw. Desc | Image

Generate Collection | Print

| Term | Documents |
|---|---|
| SIMCON | 4 |
| SIMCONS | 1 |
| SIMCON.USPT. | 4 |
| (SIMCON).USPT. | 4 |

**Display Format:** TI | Change Format

Previous Page    Next Page

| **Set Name** side by side | **Query** | **Hit Count** | **Set Name** result set |
|---|---|---|---|
| | *DB=USPT; PLUR=YES; OP=OR* | | |
| L3 | (slurry adj infiltrat$ adj mat adj concrete) | 0 | L3 |
| L2 | (cure$ adj mortar$ adj reinforc$ adj steel adj fiber adj mat) | 0 | L2 |
| L1 | simcon | 4 | L1 |

END OF SEARCH HISTORY

# WEST

Help | Logout | Interrupt

Main Menu | Search Form | Posting Counts | Show S Numbers | Edit S Numbers | Preferences | Cases

**Search Results -**

| Term | Documents |
|---|---|
| SLURRY | 121789 |
| SLURRIES | 20462 |
| SLURRYS | 160 |
| MAT | 41504 |
| MATS | 16182 |
| CONCRETE | 85804 |
| CONCRETES | 1840 |
| INFILTRAT$ | 0 |
| INFILTRAT | 2 |
| INFILTRATABILITY | 5 |
| INFILTRATABLE | 48 |
| ((SLURRY ADJ INFILTRAT$ ADJ MAT ADJ CONCRETE)).USPT. | 0 |

There are more results than shown above. Click here to view the entire set.

**Database:**

US Patents Full-Text Database
US Pre-Grant Publication Full-Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

**Search:** L3

Refine Search

Recall Text | Clear

**Search History**

**DATE: Monday, October 20, 2003** Printable Copy Create Case

WEST

Generate Collection Print

L1: Entry 1 of 4 File: USPT Jan 16, 2001

DOCUMENT-IDENTIFIER: US 6174595 B1
TITLE: Composites under self-compression

Brief Summary Text (22):
Long or continuous fibers have been used in a process called SIMCON, which is slurry infiltrated continuous fiber mat reinforced concrete. The mat systems are disclosed in U.S. Pat. No. 3,637,457, U.S. Pat. No. 4,414,262 and U.S. Pat. No. 4,617,219, where three dimensional polymeric nonwovens are used and in U.S. Pat. No. 5,571,628 where shaped mats of long steel fibers are disclosed. The slurry infiltrated mats generally provide the most improved mechanical properties of the composites in the brittle matrix composite group. The improved tensile and flexural properties result from the use of long or continuous fibers. The definition of continuous fibers depends somewhat on the specific material sets, but they can generally be defined by the aspect ratio of the fibers. The aspect ratio is the ratio of the fiber's average length to the fiber's average diameter. Fibers with aspect ratios over 500 generally act as continuous fibers while fibers with aspect ratios below about 200 exhibit composite properties that are greatly affected by fiber end effects and thus act as short fibers in composites. Brittle matrix composites are unusually sensitive to short fibers because the fiber-matrix bond is often weak and the matrix deforms by cracking, thus further weakening the bond.

Brief Summary Text (27):
The effects of application of pressure during hardening on the mechanical properties of hardened cementitious compositions are related to the reduction of voids or flaws. Voids in cementitious systems can be formed during the hardening process due to shrinkage over long time periods. In Neville, A. M., Properties of Concrete, Forth Edition, Wily, New York, 1996, pg. 435, shrinkage is described as continuing at a rapid rate for three months to one year after casting. When pastes are used which contain high concentrations of cement, such as used in SIMCON, the magnitude of the shrinkage can be very large. Therefore the application of pressure for long time periods would seem to be required in order to improve the mechanical properties of cementitious composites by consolidation. This is usually impractical in an industrial operation.

Detailed Description Text (86):
The twisted samples, 3-C and 3-D, also demonstrate the importance of the lateral support. When the samples of Example II and III were tensioned, substantial improvements in flexural properties were obtained only when the lateral support was also present. It is believed that the roll of the temporary lengthwise tension was, in part, to align the lengthwise fibers along the lengthwise axis of the composite so that the fibers could more effectively cooperate to elastically reinforce the composite, particularly in the tension zone, during bending. Further, the elastic energy that was stored as a result of the temporary tension was available to provide an elastic compressive response to deformations such as shrinkage and bending, as long as the elastic energy could act with the lateral support. It is believed that the effects of the temporary tension interacted with the lateral fiber support to create a compressive micro-environment for the matrix when the composite was challenged. This phenomenon is termed self-compression. In the prior art, brittle matrix composites such as SIMCON do not have the advantageously positioned fibers or the tension aligned fibers or the stored elastic energy. In these examples most of the energy was stored in lengthwise fibers, however, it is reasonable to believe that in other constructions, the lateral fibers could play more of a roll in

WEST

Generate Collection Print

L1: Entry 2 of 4 File: USPT Jul 14, 1998

DOCUMENT-IDENTIFIER: US 5780806 A
TITLE: Laser ablation system, and method of decontaminating surfaces

Detailed Description Text (11):
To test the efficiency of laser light ablation systems, SIMCON (simulated contamination) coupons were used as workpieces.

Detailed Description Text (12):
Two types of SIMCON coupons were used for these tests. SIMCON I coupons, which simulate "loose" or surface contamination, were prepared. The SIMCON I coupons comprised known quantities of Cs and Zr nitrate salts that were dried on the surface of one inch diameter stainless steel coupons. SIMCON II coupons, which simulate "fixed" contamination, were also prepared. The SIMCON II coupons comprised known quantities of Cs and Zr nitrate salts that were baked on one inch diameter stainless steel coupons at high temperature for a period of 24 hours. SIMCON I coupons comprise a coating that is easily removed. SIMCON II coupons, on the other hand, comprise a tenacious coating that is difficult to remove. X-ray fluorescence spectrometry was used to determine the respective amounts of Cs and Zr on the surface of the coupons both before and after laser ablation experiments were performed. These SIMCON coupons are a good representation of the types of contamination that can be found in an environment that may contain radioactive contamination.

Detailed Description Text (53):
Tests show that when SIMCON 2 coupons were cleaned using the optimum operating parameters an average of 100% of the Cs and 95.14% of the Zr simulated contaminants were removed. When SIMCON 2 Coupons were cleaned within the operating range, an average of 99.43% of the Cs and 93.32% of the Zr simulated contaminants were removed.

L1: Entry 3 of 4 File: USPT Aug 4, 1992

DOCUMENT-IDENTIFIER: US 5136528 A
TITLE: Maintenance and operational simulators

Detailed Description Text (2):
Referring now to FIG. 1, a representation of a maintenance and operational simulator 10, to simulate a complex system, such as the electronic command and control equipment for a surface-to-air missile system, is shown to include an instructor/student console 20 and a plurality of simulation control assemblies 40a-40b, (SIMCON) which are used to interface the instructor/student console 20 to a mock-up 80 of the system being simulated. Here the mock-up 80 includes a control panel (not shown) having disposed thereon switches, lamps, displays, rotational devices, such as potentiometers, track balls, wheels, as well as push buttons. The mock-up 80 further includes, as necessary, circuit breakers and dummy electronic modules, which may be reset or removed by a maintenance technician during an exercise. In the particular system simulated, a HAWK surface-to-air missile system, the missile system (not shown) includes three distinct subsystems. A continuous wave acquisition radar (CWAR) (not shown) for detection and acquisition of targets; a high power illuminator radar (HPI) (not shown) for tracking and engagement of the target; and a platoon command post 82 (PCP), which is the central control for the HAWK missile system. The HAWK missile system also includes a launcher and missile (not shown). Here the simulator 10 simulates the PCP 82. The PCP 82 includes three principle subsystems, the Data Processor (DP) 84, the Tactical Display and Engagement Console (TDEC) 86, and the Power Distribution Unit (PDU) 88. The PCP 82 includes two displays. In the particular SIMCON to be described, a SIMCON controls one display. Thus, two SIMCONs are provided to control simulation of the PCP 82. One SIMCON 40a controls DP 84 the other SIMCON 40b controls TDEC 86 with the control of the PDU being shared between the two SIMCONs 40a, 40b. Under certain circumstances, depending upon the exact configurations of the simulation 10 and requirements of the system mock-up, one SIMCON or more than two SIMCONs may be used. In any event, one or more SIMCONs may be controlled by the instructor/student console 20.

Detailed Description Text (3):
The system mock-up 82 includes an actual or substantially actual control panel (not shown) as typically used by an operator or maintainance technician. Such control panel includes inter alia switches, displays, indicator lamps, speakers, potentiometers, joy sticks, hand wheels, and the like as used on the actual system. These devices are wired as required via wire harnesses 48a, 48b to SIMCONs 40a, 40b. To the greatest extent feasible, the control panel of the system mock-up 82 duplicates the control panel of the actual operational system. However, actual circuits used in the system being simulated are not used in the system mock-up 82, thus reducing cost and complexity. When maintenance operations require the need to remove circuit boards, such boards can be simulated by blank boards mounted in the equipment but hooked up to simulate that an electrical path has been broken. This occurrence would be detected by the SIMCON, as will be described.

Detailed Description Text (4):
Either operation or maintenance of the actual system is simulated by the system simulator 10 depending upon software models of the system which are resident in the instructor/student console 20 and executed by the instructor/student console during a simulation, as will be described. Thus, suffice it here to say that the simulator 10 operates by commands sent via the instructor/student console 20 through the SIMCON modules 40a-40b. In one senario, for example, a maintenance instructor may

choose a particular maintenance routine to teach or test a student in maintenance of the system 82. The instructor accesses a library of maintenance routine software models stored in the instructor/student console 20, as will be described, and selects one of the routines. The selected routine is executed by a computer 11, which is part of the instructor/student console, causing messages to be sent to the system mock-up 82 and received from the system mock-up 82 through SIMCONs 40a-40b. Such messages simulate faults in the system mock-up 82. In accordance with a predetermined fault isolation technique as specified, for example, in a maintenance manual specific to the system mock-up 82, the student proceeds through maintenance steps to isolate and correct the simulated fault. Thus, the student is positioned at the system mock-up 82 and adjusts switches and rotational devices; removes mockup circuit boards, takes measurements, and performs other maintenance checks on the system mock-up 82. Messages from the instructor/student console 20 via lines 47a, 47b and sent to and decoded by the SIMCON 40a, 40b and in response the SIMCONs 40a, 40b sets indicator lamps, meters, and the like via wire harnesses 48a, 48b on the system mock-up 82. The SIMCONs 40a, 40b also detect and monitor actions taken by the student at the system mock-up 80. The SIMCONs 40a, 40b prepares and formats messages to send to the instructor/student console 20 indicating that such actions have occurred. The instructor/student console 20 software makes a record of such actions and may determine the appropriateness of the action taken by the student. Thus, the instructor can monitor the student's performance. Alternatively, the student may select his own routine from a predetermined list of such routines as determined by the instructor and practice the selected maintenance routine.

Detailed Description Text (6):
The student/instructor console 20, thus stores and executes maintenance model routines. Each maintenance model includes sequences of instructions which selectively determine faults to be simulated on the mock-up of the system 80. Actual simulation of such faults, however, is performed by one of, here, a plurality of SIMCONs 40a-40b. Each one of SIMCONs 40a-40b, as will be further described in conjunction with FIGS. 2 through 8, are identical and provide all the firmware and hardware necessary to simulate any combination of the aforementioned switches, indicator lamps, speakers, potentiometers, and so forth, which in general would be encountered when simulating any complex system.

Detailed Description Text (8):
The instructor/student console 20 further includes, here a plurality of serial interface modules 19a-19b, which interface the instructor/student console 20 to a corresponding one of the SIMCONs 40a-40b, as shown. Here, the instructor/student console 20 is any personal computer system, preferrable to an IBM-AT (IBM, Inc.) or AT compatible personnel computer system, operating in an MS DOS (Microsoft, Inc.) environment. Resident on the hard disk 15b of the instructor/student console 20, therefore, are software routines, which interface the instructor/student console to the SIMCONs 40a-40b, and models of maintenance or operation routines specific to the system being simulated, as will be further described in conjunction with FIG. 9.

Detailed Description Text (9):
Referring now to FIG. 2, a SIMCON 40a representative of each one of said SIMCONs 40a-40b is shown to include a central processing unit 44 and associated memory 45, here said memory including EPROM, which has resident therein firmware to interface functional modules 110-710 of the SIMCON 40a to the CPU 44 and RAM, which is loaded with program modules as required to provide the memory from which the CPU actually executes the program to interface the SIMCON to the instructor/student console 20 and mock-up 82. CPU module 44 is here a single board computer designed to operate on the IBM-AT bus. It thus uses any of the family of 80.times.86 (INTEL, Inc.) microprocessor microcircuits. The maximum real time processing requirements of the system mock-up 82 is the primary factor used in determining which of the processors to select. The CPU module in combination with the firmware stored in EPROM is used to interface the SIMCON 40a to the instructor/student console 20 (FIG. 1), as well as, the simulated system 82 (FIG. 1). Here, the SIMCON firmware is generally described in conjunction with FIG. 10. Suffice it here to say however, that the firmware resident in memory 45 is used by CPU 44 to interface the various functional modules 110-710 between the mock-up system 82 and the instructor/student console 10 of FIG. 1.

Detailed Description Text (10):
Still referring to FIG. 2, the functional modules included in a typical SIMCON are here shown to include a lamp module 110 used to control, via a harness 148a, devices on the mock-up 82, such as lamps 82a, or relays, circuit breakers, etc., as will be further described in conjunction with FIG. 3, a switch module 210 used to detect, via a harness 248a, switches 82b being opened and closed, as will be further described in conjunction with FIG. 4, and a digital/analog conversion module (D/A module) 310 used to control, via a harness 348a, analog meters 82c, as will be further described in conjunction with FIG. 5. Moreover, the SIMCON module 40 further includes an analog to digital module (A/D module) 410 used to detect, via a harness 448a, rotational motion, such as from potentiometers 82d, as will be described in conjunction with FIG. 6, a graphics processor module 510 used to control, via a harness 548a, displays 82e. Module 510 is here a commercially available module from National Designs, Inc., which uses a T.I. 34010 graphics processor on a Genesis 1024 card. The SIMCON 40a further includes an audio module 610 used to send, via a harness 648a, audio to a speaker 82f, as will be described in conjunction with FIG. 7, and a serial interface module 710 having lines 47a and 748a, as will be described in conjunction with FIG. 8.

Detailed Description Text (11):
The graphics control of a system mock-up is included in the SIMCON. This module, like the other modules in the SIMCON is substantially generic. That is, the hardware and much of the firmware used to display particular images is generic. What is here unique to the particular system being simulated 82 is that portion of the firmware which selects particular sequences of images to be displayed. This firmware is unique to the system being simulated. Here commercial RS-232 serial interfaces (FIG. 1) are used to interface the instructor/student console 20 to the serial interface module 710 in the SIMCON 40a via line 47a. This communication port operating, here at 38.4K baud is too slow to adequately control, update, and otherwise generate graphics for the displays required in the mock up of the system 82. Although the instructor/student console 20 could be directly connected, via a high speed interface, to the system mock-up 82 to provide the graphics information, this arrangement is not preferred because it would force a particular configuration for the instructor/student console 20. Thus, some of the firmware on the graphics card must be changed to simulate different systems. Here, the SIMCON is installed in a standard 19" rack mounted case (not shown) including a 12 slot IBM-AT compatible passive back plane (not shown) with associated power supply (not shown). Into this rack (not shown) are inserted combination of the abovementioned modules to perform the required hardware aspects of the generic simulation functions described above. All of the modules, accordingly, conform to the IBM-AT form fit factors for both mechanical and electrical specifications. Any mix of modules, accordingly, is possible in accordance with the required simulation tasks.

Detailed Description Text (12):
Here serial interface module 710, as will be further described in conjunction with FIG. 8, provides a serial interface along line 47a between the SIMCON 40a and a corresponding serial interface module 19a of the instructor/student console 20 (FIG. 1). Thus, suffice it here to say that CPU module 44 provides a internal bus 42 comprised of address lines 42a, data lines 42b, and control lines 42c. Each one of the functional modules 110-710 are interfaced to the CPU 44, via the bus 42.

Detailed Description Text (13):
A typical example of the simulator 10 in operation to simulate the mock-up of system 82 would include the switch module 210 (FIG. 4) monitoring switches on a switch matrix 82b (FIG. 4A) and the lamp module 210 controlling indicator lamps 82a on the system mock-up 82. The switch module 310 of SIMCON 40a will interface the instructor/student console 20 to switches on the mock-up 82 through a wiring harness, here 248a. The switches on the mock-up 82 are wired together to provide a matrix of columns and rows, as shown in FIG. 4A, and as will be further described. In such a switch matrix 82b may be included push-buttons switches, toggle switches, as well as a circuit board, which when installed properly, provides a switch indication of being a closed path and when removed provide a switch indication of an open path. Activation of a device, such as a switch or insertion of or removal of a module on the mock-up 82 results in a row/column detection by hardware and firmware resident in module 210. The SIMCON, having no specific intelligence regarding the

device being simulated or the meaning of the activation of the device, simply formats a digital word and sends it to the instructor/student console 20 via the serial interfaces indicating the row and column of the device which was detected. The message would also include information as to the state of the device, (i.e. whether a closed or open path was detected). The software resident in the instructor/student console 20 forwards these digital messages to an appropriate software model resident in the instructor/student console 20. During execution of this model software, a determination is made, as to what action if any should be taken as a result of the device being simulated. Thus, resident in the maintenance model software is the capability of determining the actual location of the device with respect to the mock-up system 82, as well as, the meaning of the operation which occurred by changing the status of the device. For example, activation of a switch may require a lamp to be turned "on" in the system mock-up 82 corresponding to the activation of the switch. Thus, the instructor/student console software will format a serial message to the SIMCON module 110 which will be fed to the SIMCON module 40a via the CPU 44 to format instructions to lamp module 110 to activate one of the indicator lamps.

Detailed Description Text (14):
Referring now to FIG. 3, a lamp module 110 which is here used to selectively activate and de-activate indicator lamps 82a, as well as, relays, circuit breakers (not shown) of the system mock-up 82 (FIG. 2) is shown to include module select logic 112, here a programmable logic array, which is fed an address via address lines on address bus 42a from SIMCON processor 44 of FIG. 2, as well as, an address enable "AEN" or valid signal from bus 42c (part of the CPU control bus), which indicates when the information on address bus 42a is valid.

Detailed Description Text (17):
This data will appear at outputs of register 134a-134dd and provide signals on individual ones of wires contained in wiring harness 148a, which is connected to lamps 82a, relays, and the like on system 82 to selectively activate or de-activate the bank of indicator lamps 82a, for example, of subsystem 82. Thus, the instructor/student console 20 will format a message to SIMCON 40a, which corresponds to an address to select one of the registers 134a-134dd on module 110 to receive the data. The instructor/student console 20 will also format the data to selectively activate or de-activate one or more of the lamps from the data in the selected register. The address is thus decoded from internal address bus ABUS 122 and produces register enables, REG.sub.0 -REG.sub.29. Data on DBUS 132 is fed to each of a plurality of registers 134a-134dd, as shown, and in accordance with the assertion of one of the register enables, REG.sub.0 -REG.sub.29 such data are loaded in parallel into the selected one of registers 134a-134dd. In this manner, the outputs of such registers are connected to the lamps 82a of the subsystem 82 and in accordance with the state of signals fed to the registers 134a-134dd selected ones of the lamps 82a are activacted or de-activated on the mock-up of the subsystem 82. Thus, with this particular module, the instructor/student console, in accordance with a predetermined senario, may be used to feed commands to the simulation control module to configure the lamp module 110 to select and control devices, such as the indicator lamps 82 a on the system mock-up 82.

Detailed Description Text (20):
Module 210 further includes address buffer 220, which when enabled by master enable, MSTEN, permits address information from bus 42a to be coupled to the internal address bus ABUS 222 of module 210, as shown. Master enable MSTEN, when asserted, also permits data to be coupled between data bus 42b and an internal data bus DBUS 232 through tranceivers 230. A read or write signal from CPU 44 (FIG. 2) determines direction of data flow on DBUS. The function of switch module 210 is to detect the occurrence of an electrical path being made or broken by the assertion or de-assertion of a switch on the simulated system 82a via harness 248a. Such an occurrence therefore, requires that an interrupt be generated, and fed back to the SIMCON central processor 44. Accordingly, here the switch module includes a switch processor 240, which generates an interrupt to the SIMCON CPU module 44 to indicate to the SIMCON CPU 44 that a switch has been asserted. The SIMCON CPU module in turn generates commands to the switch processor to determine the address, that is the row and column, of the switch which was detected, as well as the state of change of the switch. The SIMCON CPU 44 obtains access and control of internal buses ABUS, DBUS to

access memory 242, via a second programmable logic array 221, which is fed address lines from bus 42a and control signals IOW, IOR, AEN from control bus 42c and is used for bus control and permits the module CPU 240 to relinquish control of the internal buses to the SIMCON CPU 44 by feeding signals IRQ3EN, Z80EN, MSTREQ, and INTREG to CPU 240 to cause CPU 240 to relinquish control of internal buses to CPU 44. Such signals also control generation of SLAVEN to permit processor 240 to run.

Detailed Description Text (21):
The module further includes a third programmable logic array 260, here used to provide bus control for memory and coder for the switch module 210. This PLA is used to decode address bus 42a to enable memory when CPU processor 44 (FIG. 2) is accessing the internal data and address buses of the module 210, or when the switch processor is using its internal buses. When accessed by the SIMCON CPU 44 address information is on bus 42a and I/O request signals IOW, IOR are from control bus 42c. When accessed by switch processor 240, address information is on bus 222 and control signals IMREQ, IWR, IRD are provided from the switch processor 240. The I/O programmable logic array provides output signals SELRAM, SELROM, RDRAM, WRRAM, which are provided to the memories 242-244 and the bank enable signals BANK.sub.0 -BANK.sub.3, which are fed to the decoders 224a-224d.

Detailed Description Text (22):
Firmware which controls the switch processor 240 is stored in EPROM memory 244. Memory 242 is here, a random access memory, which is used by the switch processor 240 in processing the information from the SIMCON processor, as well as from the switch matrix 82b (FIG. 4A). The switch processor in conjuntion with firmware stored in memory 244 polls each one of the switches in the switch matrix 82b by send strobes sequentially along each of the column lines and detecting the presence or absence of the strobes along the row lines. Thus, the switch processor formulates address messages, which are fed, via ABUS 222, to column decoders 224a-224d which select one of here of 64 columns by asserting a logic level "0" on the selected column line COL 1 to COL 64. Signals BANK.sub.0 -BANK.sub.3 are fed to the decoders to enable the proper decoder and to serve as an indication that address information on ABUS 222 is intended as coded column addresses. Here column decoders are 74LS174 types 4 to 16 line decoders. As shown in FIG. 4A, each column has eight switches having one terminal thereof connected together via a column line. The other terminal of each switch in each one of the 64 columns (2 columns shown) is connected to a diode D for isolation. These terminals are connected together via a common row line, here one of ROW 1 to ROW 8 lines.

Detailed Description Text (25):
Thus, the state of each switch in each COL 1 is fed onto the bus 232 and is fed by the processor into memory 242 and the processor makes the determination as to whether the state of any one of the switches in the bus has changed, and if such state has changed, will generate an interrupt to the SIMCON processor to cause the processor 240 to halt while the SIMCON processor 44 reads from memory 242 the status of all of the switches. Due to the speed at which the CPU processor runs and problems of contact bounce, the switch matrix 82b is generally scanned at a 40 millisecond rate and a 1 to 3 millisecond hold or wait state is provided while each switch is scanned by having BANKSEL feed a monostable multivibrator 249, whose output is fed to the wait state terminal on the Z80 microprocessor 440. The multivibrator is used to provide a wait state pulse of typically 1 to 3 milliseconds in duration.

Detailed Description Text (28):
Thus, master enable MSTEN is fed to line receivers 320 to couple address information on bus 42a to internal address ABUS 322, as well as, to transceivers 330 to couple data information on data bus 42b to internal data DBUS 332 as was generally described for module 110. The address information coupled to address ABUS 322 (Bits A03-A07) is fed to a 3-8 decoder 324a of a 74LS138 type, which provides a plurality of, here 8, output enables. Here such enables BLK.sub.0 -BLK.sub.7 are D/A block enables, which are used to enable one of eight additional 3-8 decoders 326a-326g, which also decode address lines A00-A02, as shown. Thus, decoders 324a-324g produce enables E.sub.0 -E.sub.63 and are fed to respective D/A converters DAC.sub.0 -DAC.sub.63. Thus, each of said 8 block decoders 324a-324g decodes 8 different lines and accordingly 64 unique enable outputs are provided from the decoder circuits.

Such enables E.sub.0 -E.sub.63 are fed to respective one of digital to analog converters, here DAC-6088 type converters. While one of enables E.sub.0 -E.sub.63 is selectively valid, data is fed on internal data bus 332 and such data is fed to each one of the digital to analog converters DAC-1-DAC-63. A hardware generated wait state using a 74123 type multivibrator circuit 342 is provided back to the SIMCON processor to permit the D/A converters to update properly.

Detailed Description Text (30):
Thus, the instructor/student console provides selected formatted messages to the SIMCON 40a. The SIMCON processor 42, directs such messages to the D/A module 310. Such messages correspond to address messages, which select the one of the analog meters 82c of the simulated system 82, as well as a voltage level to feed to the meter to cause a selected level to be displayed by the selected meter 82c. Such meter 82c is thus selected by activating one of the enable outputs E.sub.0 -E.sub.63 to activate a corresponding one of digital to analog converters DAC-1-DAC-63 while feeding the predetermined digital data to the digital to analog converter to provide in response a predetermined analog output on one of the output lines AOUT.sub.0 -AOUT.sub.63 of harness 348a.

Detailed Description Text (32):
Thus, the analog to digital module 410 is shown to include module select logic 412, here also a programmable logic array, which is fed address lines A12-A19 via address bus 42a address enable AEN a generally described in conjunction with FIG. 3 and signal SLAVEN from A/D module processor 440. Here said module select logic 412 provides MSTEN a master enable corresponding to the SIMCON writing to the A/D module 10 with the module under control of an A/D module processor 440. Here the processor is a Z80 microprocessor from Zilog, Inc. As with the switch module 210 (FIG. 4), the A/D module processor 440 will relinquish control of the buses in the module to the SIMCON CPU 42 when requested. Here PLA 412 is active only when the A/D module processor 440 has not relinquished control of internal data buses and address buses to CPU 42. Module address logic 412 produces master enable MSTEN, which is here used to activate the line buffers 420 to couple address bus 42a (FIG. 2) to internal address ABUS 422, as well as, to enable transceivers 430 to couple data on data bus 42b to the internal data DBUS 432.

Detailed Description Text (34):
Here the analog to digital module has a separate microprocessor and control logic, as well as resident memory, which are used to control the analog to digital module 410, and thus provide a level of control below the CPU module 44 of the SIMCON. It is envisioned that the A/D conversions may generate a relatively large number of interrupts, as well as noise, as a potentiometer or track ball is adjusted. Most of these adjustments will be relatively small compared to the time scale of the microprocessor 44 (FIG. 2). Thus, here the CPU 440 is provided to permit threshold levels to be independently set for each channel. The CPU 440 thus determines whether the converted information from a channel represent a significant change in adjustment of potentiometers, track balls, and the like, above the present threshold, and thus significant enough to generate an interrupt to the CPU module 44 of the SIMCON 40a. The sensitivity or magnitude of change which the analog to digital module will use to initiate an interrupt sequence to the instructor/student console 20 is selectively initialized for each channel by an initialization sequence from the instructor/student console 20.

Detailed Description Text (38):
Using firmware routine stored in memory 444a, the value of the data word, converter, and channel will be ascertained to determine whether or not a difference in the value of the data and a previous value (i.e. two potentiometers settings), for example, is beyond the threshold for the channel to cause the CPU 440 to generate an interrupt to the SIMCON CPU 42. Thus, if the interrupt is generated by the CPU 440 of analog to digital module 410, the digital representation of the value of the analog signal converted, as well as, the address which corresponds to the particular channel converted, will be read by the SIMCON CPU 42 from the memory 444a. A second programmable logic array 428 is fed address information from bus 42a, as well as I/O signals IOW and IOR from control bus 42c and an address enable. This PLA 428 generates output signals used to set and clear interrupts on the processor 440, inhibit the processor 440, thus obtain access to the internal buses of the module

440. This scenario is used to set threshold levels, as well as read converted values for each of the channels. Signals Z80EN and MSTREQ are used to disable the processor 440 and generates SLAVEN to inhibit the module 410 from interacting with the CPU 440. IRQ3EN and INTREG are used to indicate to the SIMCON CPU processor 44 that an interrupt from the 440 processor has been asserted.

Detailed Description Text (50):
A typical address format for serial I/F module 710 is: ##STR6## Master enable MSTEN, in conjunction with IOR, IOEN, is used to activate transceivers 730. The module control logic 712 also generates enables on lines 712a-712f to selectively feed parallel information from DBUS 732 into RS-232 interfaces 734a-734f, here 16550 types in accordance with addresses fed to module control logic 712 on address bus 42a (SIMCON address bus). Thus, enables 712a-712f are provided at the output of module 712 to selectively enable one of the RS-232 interfaces 734a-734f to send or receive data from DBUS 732. Here address bits A0-A2 are used to identify data, handshaking and other information such as baud rate for the serial interface circuits 734a-734g. Each one of said interfaces is used to convert parallel information fed on data lines D0-D8 into serial, formatted information. For example, one of said channels is used to communicate with the instructor/student console 20 (FIG. 1) and thus provide information between the instructor/student console and the SIMCON 40a. Each serial interface channel includes line receivers 735a and line drivers 736a. Line drivers are here typically MC-1489 type circuits, whereas line receivers are here MC-1488 type circuits.

Detailed Description Text (52):
Referring now to FIG. 9, a software tree diagram 60 representative of a typical software package installed on the instructor/student console 20 is shown to include various software modules. The task scheduler software module 62 permits the system 10 to have multi-tasking capabilities. Multi-tasking allows the instructor/student software to run multiple SIMCONs, here 40a-40b. In the simulator, to simulate a HAWK surface-to-air missile system, the instructor/student console 20, controls the Automatic Data Processor SIMCON, and the Tactical Display and Engagement Control Console SIMCON. The task scheduler software under which modules 62a-62c operate includes an array of pointers or a queue and an array of software timers. Here the task queue is not prioritized. Any task once started will run to completion. As tasks complete they remove themselves from the the queue.

Detailed Description Text (60):
The interrupt processor software module 63 handles all serial port interrupts for the instructor/student console 20. The interrupt handler 63 here manages up to 6 serial ports, which are used to transfer data to and from the simulation control modules 40a-40b and the instructor/student console 20. Interrupts are used by the input/output devices of the SIMCON 40a-40b when the SIMCON need to be serviced by the computer 11 in the instructor/student console 20.

Detailed Description Text (75):
Simulator Diagnostics--Functions to test the SIMCON and 3D elements of the simulator.

Detailed Description Text (80):
The software diagnostics module 68 is used to verify operation of simulator hardware. Checks are provided for the instructor/student console simulator control serial link, the serial ports in the SIMCON 40a-40c as well as the lamp module 110, switch module 210, D/A module 310, A/D module 410, graphics module 510, audio module 610 and serial line module 710.

Detailed Description Text (82):
For each set of tests, the instructor/student console SIMCON serial link is always tested first. This is done as all commands and reports are sent via this link. The other tests include:

Detailed Description Text (93):
The commo processing system handles communication between the instructor/student console and SIMCON. Messages are passed between the two systems in defined formats. In the case of instructor/student console to SIMCON messages, the commo processor

formats the message and sends it on to the SIMCON via the interrupt handler. For SIMCON to instructor/student console messages, the instructor/student console commo processor decodes the message from the SIMCON and routes it to the proper handler.

Detailed Description Text (94):
The message formats for ISC - SIMCON are:

Detailed Description Text (95):
The lamp 1 and 2 messages are used to turn lamps on and off at the system mock-up (as well as relays and other items that require switching). These are provisions for two lamp modules in the SIMCON. The register number is actually the column number at the SIMCON, while the 8 bits in the lamp data value correspond to rows 0-7. Using this method allows control of 8 lamps with 1 message, reducing traffic on the serial link.

Detailed Description Text (96):
The SIMCON util message is used for general purpose messages, like commanding the SIMCON to execute diagnostics.

Detailed Description Text (97):
The format for the SIMCON util message is 5 bits for function number (32 functions total, with the second byte used for any supplementary data needed by the defined function. This message may also be used to set A/D resolutions.

Detailed Description Text (98):
The D/A messages are used to send analog information to the system mock-up 82 via D/A module 310 in the SIMCON. Channel can be a value of 0-31, while the data can be 0 (minimum reading) to 255 (max reading). This data is latched in D/A module 310 of the SIMCON.

Detailed Description Text (99):
The Com channel is used for controlling the serial ports on the SIMCON. The SIMCON will always initialize port 0, as it is the port used to communicate with the instructor/student console. This message allows initialization of the other 5 ports, as well as sending characters out any of the ports. The channel number can be any port 1-5. If the I bit is set to 1, it flags this as a command to initialize the port at the SIMCON. In this case the data contains the information for baud, parity, stop and data bits. The H bit, if set during an initial call, tells the SIMCON to ignore hardware handshaking for this port. If the I bit is low, this message sends the value in data out the port number defined in channel number.

Detailed Description Text (100):
The graphic message is used to call graphics routine in the SIMCON. The different types of routines includes point objects, lines, screens, doppler, text, and targets.

Detailed Description Text (102):
Nevertheless, a general approach to provide a software model to emulate a system is that a boolean expression of the system to be simulated is first generated. Thus, for a maintenance simulation, the model should be that of the system operating properly without any faults. Then after such a system is correctly modeled, addition submodules (not shown) are written which create faults in the model of the system without faults. These model faults result in messages which are sent to the SIMCONs 40a-40b and result in changes at the system mock-up 82. The maintenance trainee then responds to such changes, and his actions are monitored and recorded in the instructor/student console.

Detailed Description Text (103):
Referring now to FIG. 10, the firmware/package 70 used to operate a SIMCON, here SIMCON 40a (FIG. 2) resident in EPROM 45 (FIG. 2), is shown to include a task scheduler 72, which is similar to the task scheduler 62 (FIG. 9) in the instructor/student software 60 (FIG. 5). The significant difference between this task scheduler 72 and the aforementioned task scheduler is that the SIMCON task scheduler 72 runs continuously, whereas the instructor/student task scheduler 62 runs only when a scenario (i.e. module 62a-62c) is in progress. An interrupt

processor 73, as in the instructor/student console software 60, uses interrupts to signal the CPU 44 (FIG. 2) that some change in status has occurred. The SIMCON interrupt processor 73 handles these interrupts in a similar manner as the instructor/student version. Because of the assorted duties of a SIMCON, such as controlling lamps, switches, A/D channels, D/A channels, etc., there are additional interrupts which must be serviced by the SIMCON interrupt processor software 73. The library software 74 is the same library as resident on the instructor/student console.

Detailed Description Text (104):
A diagnostics module 72a, which runs under the control of task scheduler 72 is used to perform confidence tests on the SIMCON installed module cards. Test results are sent to the instructor/student console. The SIMCON diagnostics are also linked to the instructor/student console. The instructor/student console diagnostics use some of the SIMCON diagnostics routines to perform instructor/student generated diagnostic requests. The principle checks performed by the SIMCON diagnostics are to determine the card complement in the SIMCON module and perform memory and input/output checks on the installed cards.

Detailed Description Text (105):
A lamp control software module 72b, also under control of the task scheduler as are the remaining modules to be described, receives commands from the instructor/student console 20 (FIG. 1) to illuminate or extinguish a desired lamp on the system mock-up 80. Each message is decoded to determined the appropriate lamp or lamps to control. A single message can contain the information necessary to change the status of up to 8 lamps. The decoded information is passed through the lamp card where it is sent out to the individual lamps, as generally described in conjunction with FIG. 3. The code written in Microsoft, Inc. "C" is found as part of the SIMCON. C software in Appendix A.

Detailed Description Text (106):
The switch input software 72c receives interrupts generated by the CPU 240 (FIG. 4) based on firmware resident on the switch control card 210 (FIG. 3) in the SIMCON. Once an interrupt is received, the switch interrupt software accesses a formatted message from the switch card 210 (FIG. 4) identifying row, column, and switch data and sends this information to the instructor/student console computer 11 (FIG. 1) via the serial card 710. The software for the input switch modules is in Appendix B. The firmware, which is resident on the switch module, is in Appendix C.

Detailed Description Text (107):
The A/D control software in the SIMCON receives interrupts from the A/D module based firmware in the SIMCON. When an interrupt is received, formatted messages from the A/D card are sent to the communications processor for ultimate passage to the instructor/student console. The software for A/D control is found in Appendix D. The code for A/D firmware is found in Appendix E.

Detailed Description Text (108):
The D/A control software 72e receives messages from the instructor/student console (part of SIMCON, C in Appendix A) to control analog devices, such as meters on the system mock-up 80. Once received, the SIMCON decodes the messages to determine the channel and value to send to the selected meter on the mock-up system 82.

Detailed Description Text (109):
The IND1 graphics software 72f contains graphic screens and character generation routines for display at a first graphics indicator on the system mock-up 82e. This software module is generally unique to the particular system being simulated. Here, the IND1 graphics software is used to control a first display on the simulated system. IND2 graphics is used to generate screens and characters for a second display on the system 82. Generally only one of said packages are executed by a SIMCON. Both packages are installed on each SIMCON to maintain commonality.

Detailed Description Text (110):
The SIMCON como processor 72g passes messages between a SIMCON 40a and the instructor/student console 20. The como software decodes messages coming from the instructor/student console and decodes outgoing messages to the instructor/student

console. The software for the commo processor is contained in Appendix F.

Detailed Description Text (111):
The message formats for SIMCON to instructor/student console are:

Detailed Description Text (114):
The Com function simply sends data received on any of the serial channels up to the instructor/student console. Channel 0 is not reported, as that is the SIMCON instructor/student console commo channel.

Detailed Description Text (115):
The audio control module software 72i (part of SIMCON, C) controls sound generation at the instructor/student console. The audio is used to simulate an audio test or indication as would occur in an actual system. It permits the selection of the volume, frequency, as well as waveform of the desired audio information. It may also be used in operator training for example, to simulate audio dopler frequency shift.

Detailed Description Paragraph Table (2):
Byte 1 Byte 2
Lamp 1 000 register (5 bits) lamp data Lamp 2 001 register (5 bits) lamp data SIMCON Util 010 function (5 bits) data Not assigned 011 D/A card 1 100 channel # (5 bits) data D/A card 2 101 channel # (5 bits) data COM 110 #I channel # (3 bits) data Graphic 111 Format depends on message

L1: Entry 4 of 4 File: USPT May 5, 1992

DOCUMENT-IDENTIFIER: US 5111413 A
TITLE: Computer-aided engineering

Detailed Description Text (27):
The simulation kernel 10 contains basic event queuing functions used by the simulator 9, as well as code which communicates with simulation control portion (SIMCON) 12. SIMCON 12, along with simulator 9 together act as a simulation application. Separation of the simulator 9 and SIMCON 12 into separate processes greatly improves performance of the simulator because simulation need not be slowed down through interaction with the user. Incremental changes from the schematic editor (SView) 4 to the netlist manager 11, and results of the changes, are communicated via IPC (Inter-Process Communication) channel 16. IPC channel 17 is used to communicate stimulus, breakpoints, and other controls between the simulator 9 and the simulation control 12.

Detailed Description Text (28):
SIMCON 12 is connected to results display (RESDISP) 13. Both RESDISP 13 and SIMCON 12 (as well as other portions of the simulation system) have access to GC 8 and NLM 9 since both are in memory, i.e., shared memory 14 which is local to the simulator.

Detailed Description Text (29):
Memory 14 is shared between two processes, SIMCON/RESDISP and NLM/GC/KERNEL. Data structures containing hierarchical design information are placed in this memory, and a name server module of NLM provides functional access to design hierarchy by both applications. Read/write access from/to the shared memory is provided to NLM via data line 29.

Detailed Description Text (30):
RESDISP 13 and SIMCON 12 form a separate application. SIMCON 12 allows the user to interactively control the simulator, to provide stimulus, set breakpoints, and examine internal values in the simulated design. RESDISP 13 allows the user to see simulation results as waveforms or in ASCII tables. These results can be from the current simulation, as controlled by SIMCON, or the archived results from a previous simulation. Hierarchical design information is provided to RESDISP (via a name server routine-NSRLIB) via line 30 and to SIMCON via line 31.

Detailed Description Text (66):
Returning to FIG. 1, it is seen that the NLM module 11 is linked with the kernel 10 and dynamically linked with generated code 8 to form the simulator 9. The SIMCON program 12 is an interactive program used to control the simulator, providing stimulus, breakpointing, tracing, and other control functions. SIMCON 12 and the simulator 9 communicate control and the reporting of simulation results via an Inter-Process Communication (IPC) package.

Detailed Description Paragraph Table (2):
TABLE 2 ______________________________________ SIMULATION --CONTROL IS READY. THE MAXIMUM SIMULATION TIME SPECIFIED HAS BEEN REACHED. HALTING AT TIME 1000 NS, AFTER 176 DELTA TIMEPOINTS HAVE BEEN SIMULATED. INCREMENTAL UPDATE HAS COMPLETED. ALL PREVIOUS CHECKPOINTS FOR THIS CIRCUIT ARE NOW INVALID. THE MAXIMUM SIMULATION TIME SPECIFIED HAS BEEN REACHED. HALTING AT TIME 2000 NS, AFTER 382 DELTA TIMEPOINTS HAVE BEEN SIMULATED. INCREMENTAL UPDATE HAS COMPLETED. ALL PREVIOUS CHECKPOINTS FOR THIS

CIRCUIT ARE NOW INVALID. ** ASSERTION WARNING: COMPARE1: SKEW TIME IS TOO GREAT. (SCN/SIM) ** OCCURRED IN /KSERA/SERA/COMP000017 AT TIME 2305 NS AFTER 469 DELTA TIMEPOINTS HAD BEEN SIMULATED. THE MAXIMUM SIMULATION TIME SPECIFIED HAS BEEN REACHED. HALTING AT TIME 3000 NS, AFTER 587 DELTA TIMEPOINTS HAVE BEEN SIMULATED. PLOT OUTPUT WRITTEN TO "SIMCON. FIG11.PS". INCREMENTAL UPDATE HAS COMPLETED. ALL PREVIOUS CHECKPOINTS FOR THIS CIRCUIT ARE NOW INVALID. THE MAXIMUM SIMULATION TIME SPECIFIED HAS BEEN REACHED. HALTING AT TIME D4000 NS, AFTER 790 DELTA TIMEPOINTS HAVE BEEN SIMULATED.